Exhibit B

Azure Monitor overview

6/5/2020 • 9 minutes to read • Edit Online

Azure Monitor maximizes the availability and performance of your applications and services by delivering a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. It helps you understand how your applications are performing and proactively identifies issues affecting them and the resources they depend on.

Just a few examples of what you can do with Azure Monitor include:

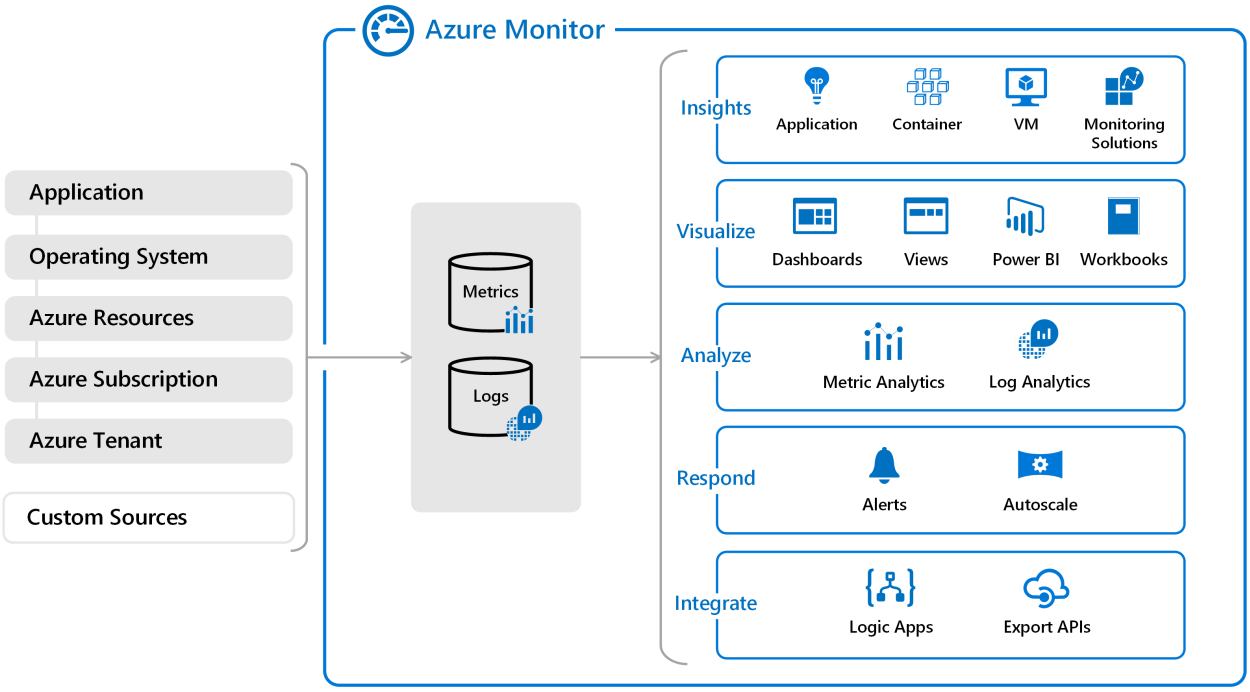
- Detect and diagnose issues across applications and dependencies with Application Insights.
- Correlate infrastructure issues with Azure Monitor for VMs and Azure Monitor for Containers.
- Drill into your monitoring data with Log Analytics for troubleshooting and deep diagnostics.
- Support operations at scale with smart alerts and automated actions.
- Create visualizations with Azure dashboards and workbooks.

NOTE

This service supports Azure delegated resource management, which lets service providers sign in to their own tenant to manage subscriptions and resource groups that customers have delegated. For more info, see Azure Lighthouse.

Overview

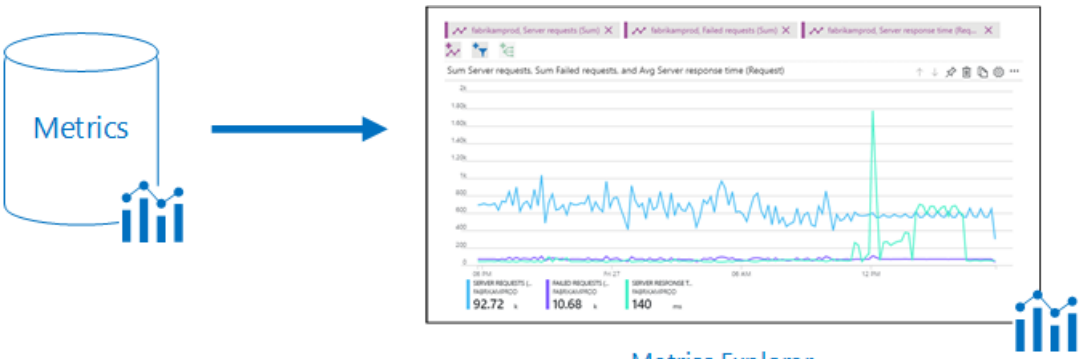
The following diagram gives a high-level view of Azure Monitor. At the center of the diagram are the data stores for metrics and logs, which are the two fundamental types of data used by Azure Monitor. On the left are the sources of monitoring data that populate these data stores. On the right are the different functions that Azure Monitor performs with this collected data such as analysis, alerting, and streaming to external systems.



Monitoring data platform

All data collected by Azure Monitor fits into one of two fundamental types, metrics and logs. Metrics are numerical values that describe some aspect of a system at a particular point in time. They are lightweight and capable of supporting near real-time scenarios. Logs contain different kinds of data organized into records with different sets of properties for each type. Telemetry such as events and traces are stored as logs in addition to performance data so that it can all be combined for analysis.

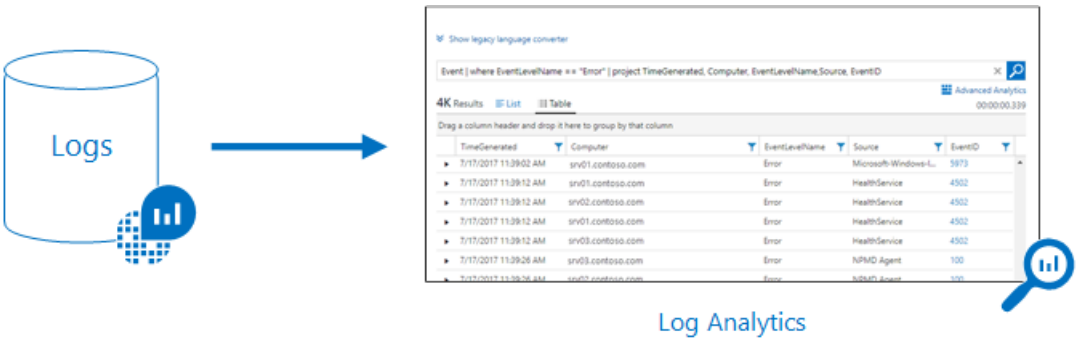
For many Azure resources, you'll see data collected by Azure Monitor right in their Overview page in the Azure portal. Have a look at any virtual machine for example, and you'll see several charts displaying performance metrics. Click on any of the graphs to open the data in metrics explorer in the Azure portal, which allows you to chart the values of multiple metrics over time. You can view the charts interactively or pin them to a dashboard to view them with other visualizations.



Metrics Explorer

Log data collected by Azure Monitor can be analyzed with queries to quickly retrieve, consolidate, and analyze collected data. You can create and test queries using Log Analytics in the Azure portal and then either directly analyze the data using these tools or save queries for use with visualizations or alert rules.

Azure Monitor uses a version of the Kusto query language used by Azure Data Explorer that is suitable for simple log queries but also includes advanced functionality such as aggregations, joins, and smart analytics. You can quickly learn the query language using multiple lessons. Particular guidance is provided to users who are already familiar with SQL and Splunk.



Log Analytics

What data does Azure Monitor collect?

Azure Monitor can collect data from a variety of sources. You can think of monitoring data for your applications in tiers ranging from your application, any operating system and services it relies on, down to the platform itself. Azure Monitor collects data from each of the following tiers:

- **Application monitoring data**: Data about the performance and functionality of the code you have written, regardless of its platform.
- **Guest OS monitoring data**: Data about the operating system on which your application is running. This could be running in Azure, another cloud, or on-premises.
- **Azure resource monitoring data**: Data about the operation of an Azure resource.
- **Azure subscription monitoring data**: Data about the operation and management of an Azure subscription, as well as data about the health and operation of Azure itself.

- **Azure tenant monitoring data**: Data about the operation of tenant-level Azure services, such as Azure Active Directory.

As soon as you create an Azure subscription and start adding resources such as virtual machines and web apps, Azure Monitor starts collecting data. Activity logs record when resources are created or modified. Metrics tell you how the resource is performing and the resources that it's consuming.

Extend the data you're collecting into the actual operation of the resources by enabling diagnostics and adding an agent to compute resources. This will collect telemetry for the internal operation of the resource and allow you to configure different data sources to collect logs and metrics from Windows and Linux guest operating system.

Enable monitoring for your App Services application or VM and virtual machine scale set application, to enable Application Insights to collect detailed information about your application including page views, application requests, and exceptions. Further verify the availability of your application by configuring an availability test to simulate user traffic.

Custom sources

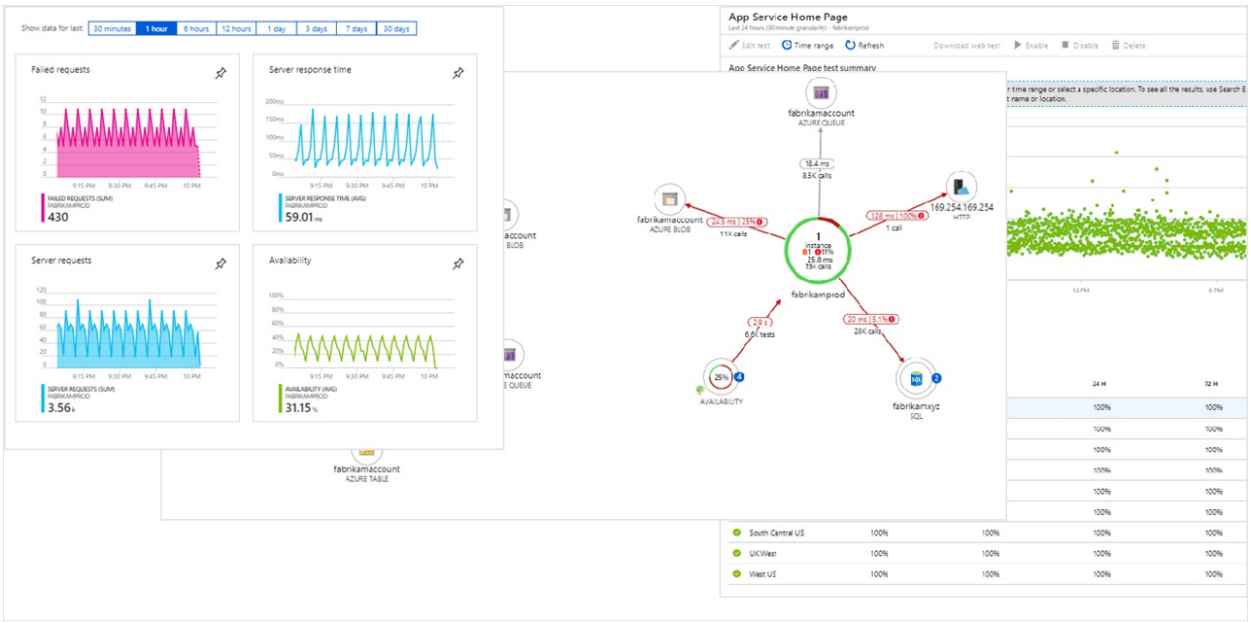
Azure Monitor can collect log data from any REST client using the Data Collector API. This allows you to create custom monitoring scenarios and extend monitoring to resources that don't expose telemetry through other sources.

Insights

Monitoring data is only useful if it can increase your visibility into the operation of your computing environment. Azure Monitor includes several features and tools that provide valuable insights into your applications and other resources that they depend on. Monitoring solutions and features such as Application Insights and Azure Monitor for containers provide deep insights into different aspects of your application and specific Azure services.

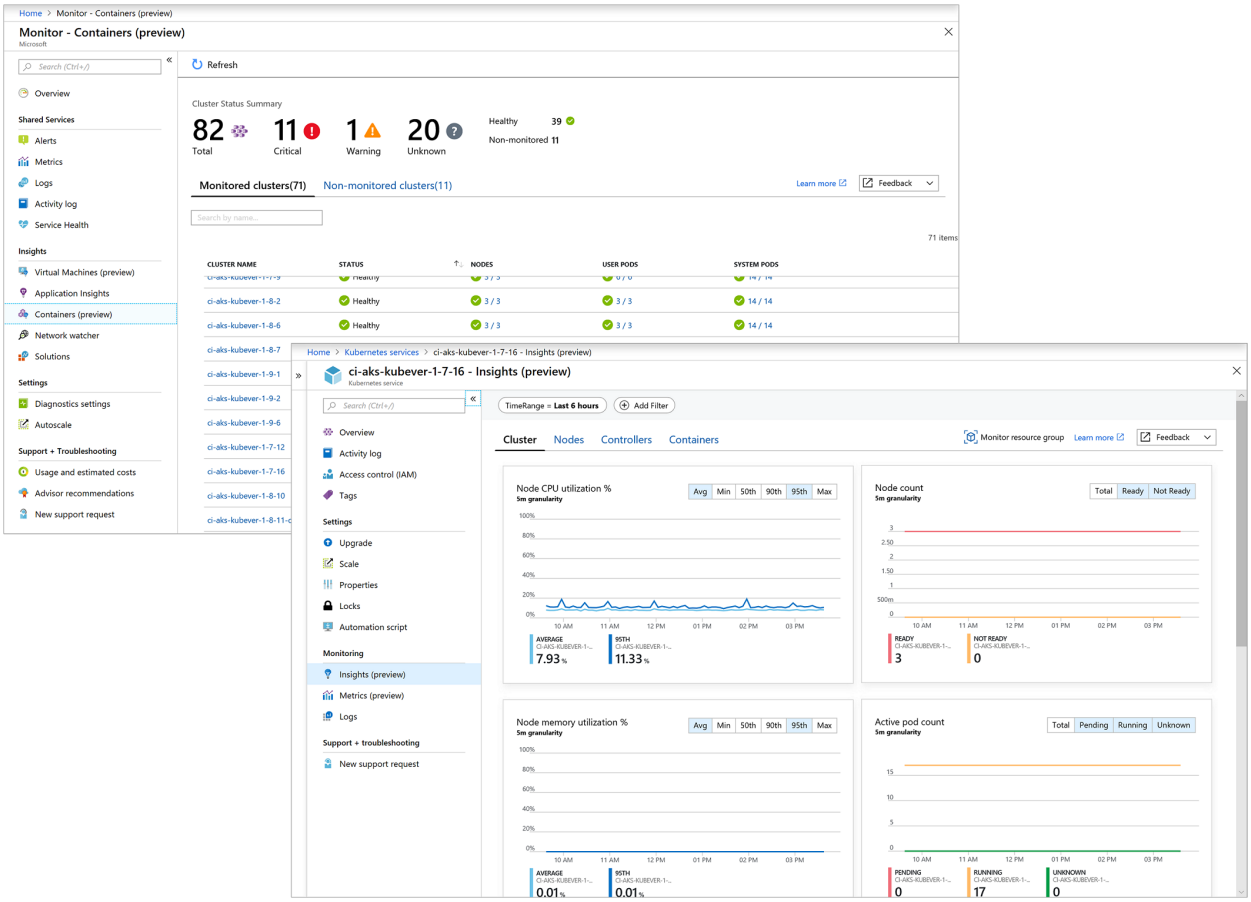
Application Insights

Application Insights monitors the availability, performance, and usage of your web applications whether they're hosted in the cloud or on-premises. It leverages the powerful data analysis platform in Azure Monitor to provide you with deep insights into your application's operations and diagnose errors without waiting for a user to report them. Application Insights includes connection points to a variety of development tools and integrates with Visual Studio to support your DevOps processes.



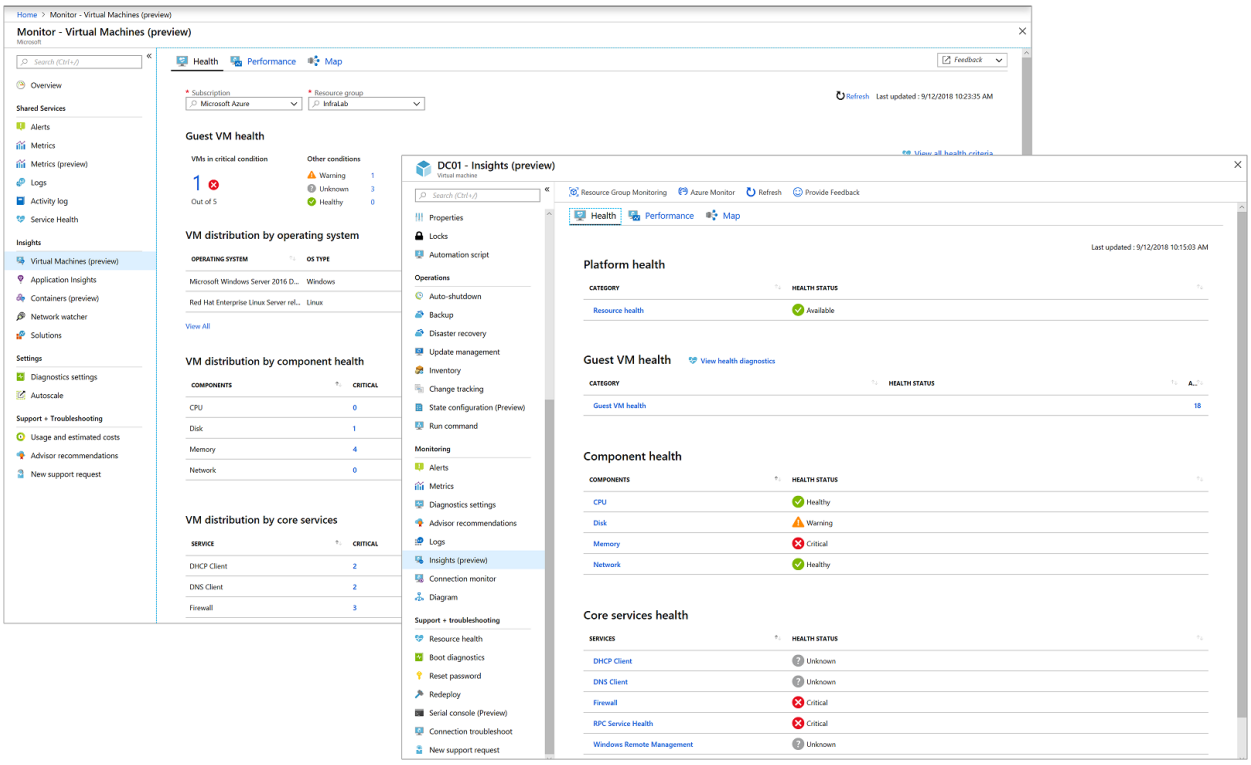
Azure Monitor for containers

Azure Monitor for containers is a feature designed to monitor the performance of container workloads deployed to managed Kubernetes clusters hosted on Azure Kubernetes Service (AKS). It gives you performance visibility by collecting memory and processor metrics from controllers, nodes, and containers that are available in Kubernetes through the Metrics API. Container logs are also collected. After you enable monitoring from Kubernetes clusters, these metrics and logs are automatically collected for you through a containerized version of the Log Analytics agent for Linux.



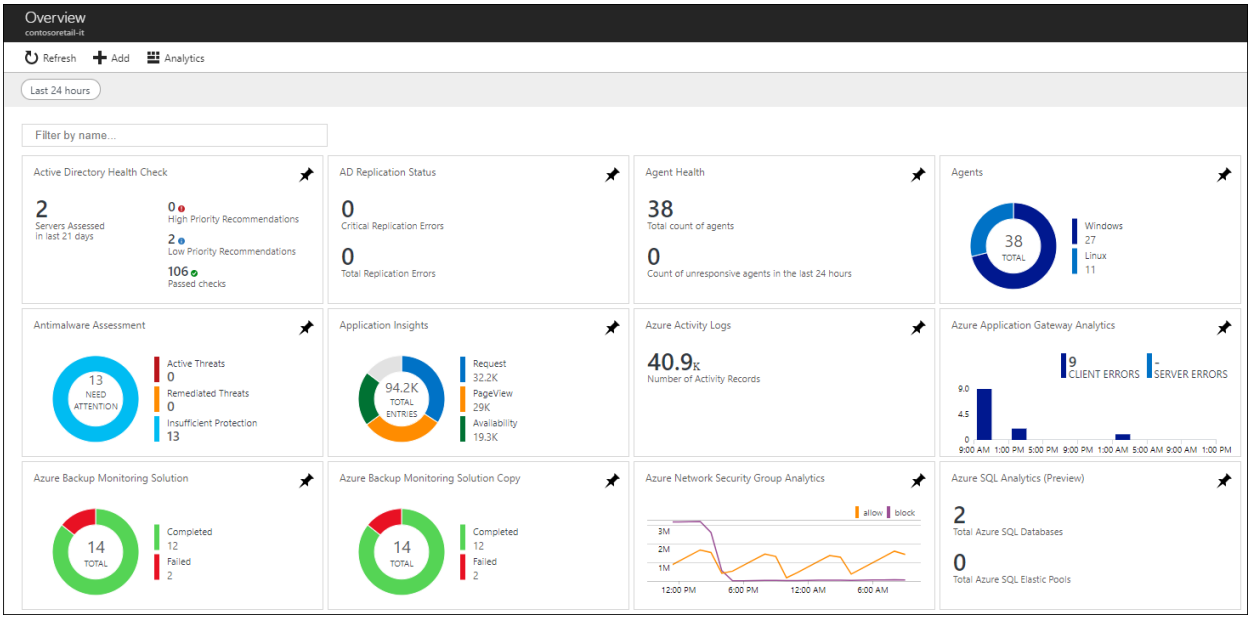
Azure Monitor for VMs

Azure Monitor for VMs monitors your Azure virtual machines (VM) at scale by analyzing the performance and health of your Windows and Linux VMs, including their different processes and interconnected dependencies on other resources and external processes. The solution includes support for monitoring performance and application dependencies for VMs hosted on-premises or another cloud provider.



Monitoring solutions

Monitoring solutions in Azure Monitor are packaged sets of logic that provide insights for a particular application or service. They include logic for collecting monitoring data for the application or service, queries to analyze that data, and views for visualization. Monitoring solutions are available from Microsoft and partners to provide monitoring for various Azure services and other applications.



Responding to critical situations

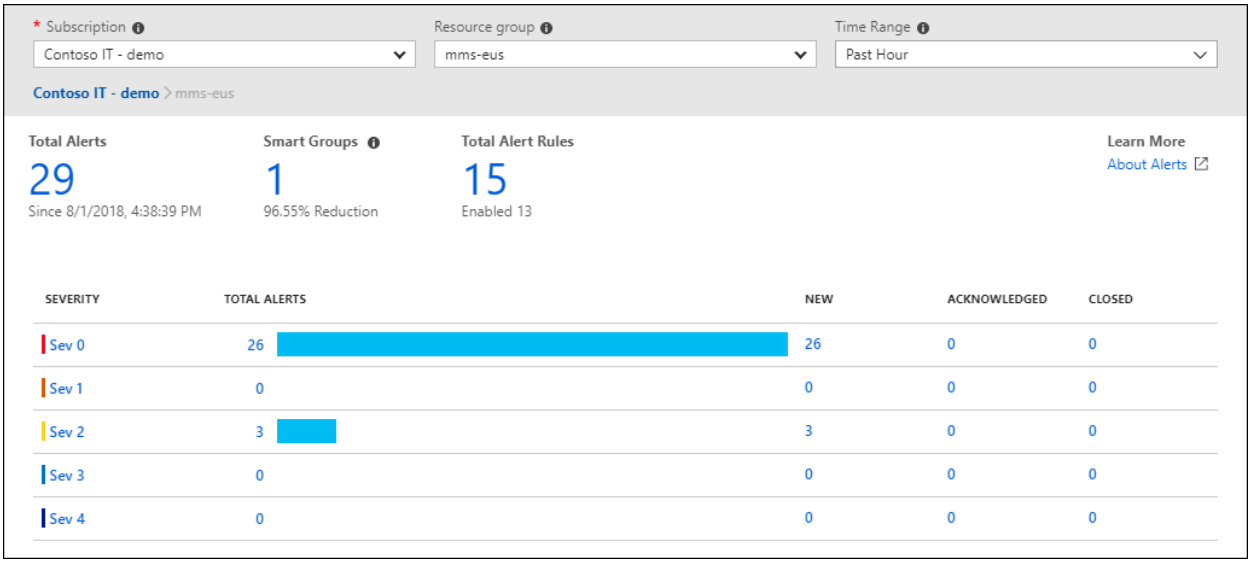
In addition to allowing you to interactively analyze monitoring data, an effective monitoring solution must be able to proactively respond to critical conditions identified in the data that it collects. This could be sending a text or mail to an administrator responsible for investigating an issue. Or you could launch an automated process that attempts to correct an error condition.

Alerts

Alerts in Azure Monitor proactively notify you of critical conditions and potentially attempt to take corrective action. Alert rules based on metrics provide near real time alerting based on numeric values, while rules based

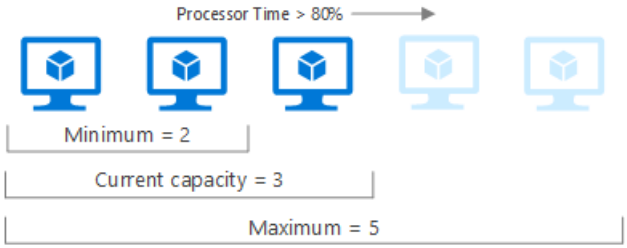
on logs allow for complex logic across data from multiple sources.

Alert rules in Azure Monitor use action groups, which contain unique sets of recipients and actions that can be shared across multiple rules. Based on your requirements, action groups can perform such actions as using webhooks to have alerts start external actions or to integrate with your ITSM tools.



Autoscale

Autoscale allows you to have the right amount of resources running to handle the load on your application. It allows you to create rules that use metrics collected by Azure Monitor to determine when to automatically add resources to handle increases in load and also save money by removing resources that are sitting idle. You specify a minimum and maximum number of instances and the logic for when to increase or decrease resources.

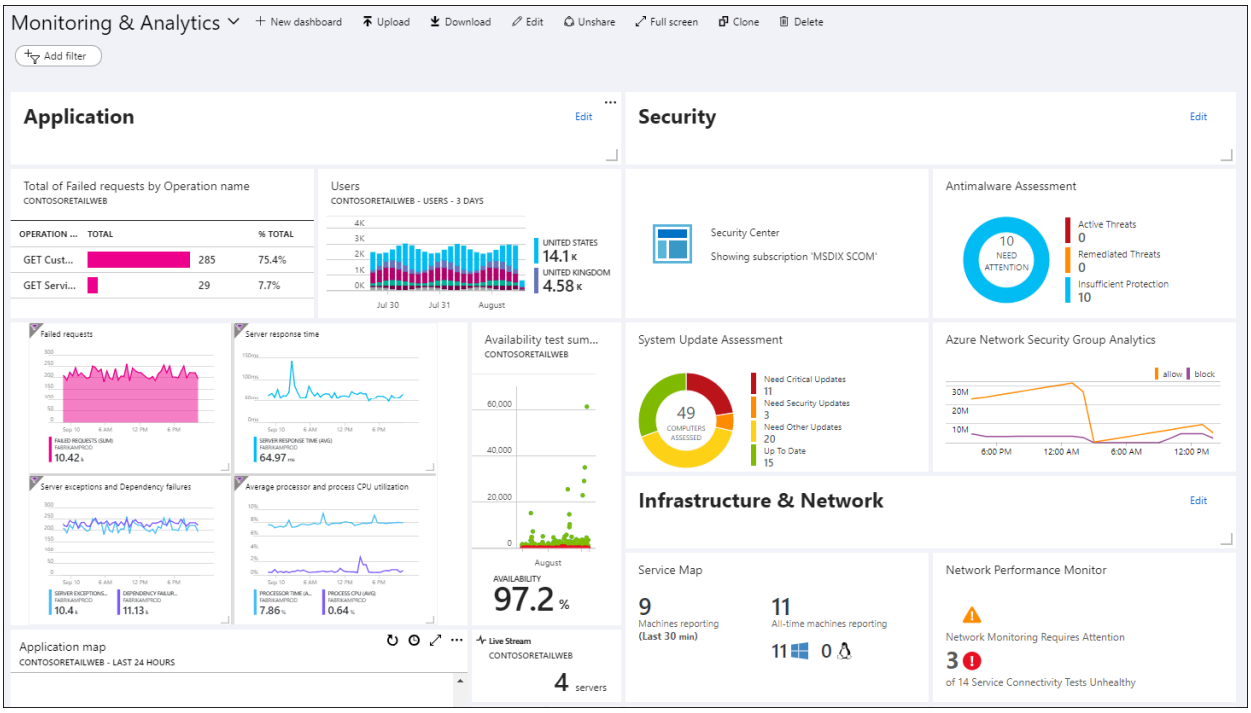


Visualizing monitoring data

Visualizations such as charts and tables are effective tools for summarizing monitoring data and presenting it to different audiences. Azure Monitor has its own features for visualizing monitoring data and leverages other Azure services for publishing it to different audiences.

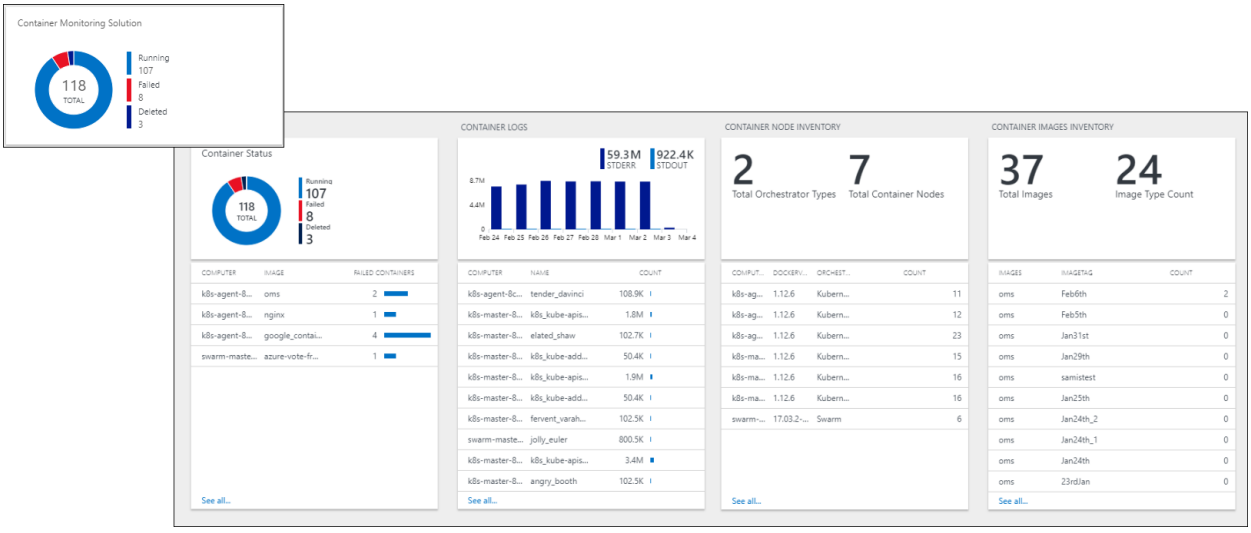
Dashboards

Azure dashboards allow you to combine different kinds of data, including both metrics and logs, into a single pane in the Azure portal. You can optionally share the dashboard with other Azure users. Elements throughout Azure Monitor can be added to an Azure dashboard in addition to the output of any log query or metrics chart. For example, you could create a dashboard that combines tiles that show a graph of metrics, a table of activity logs, a usage chart from Application Insights, and the output of a log query.



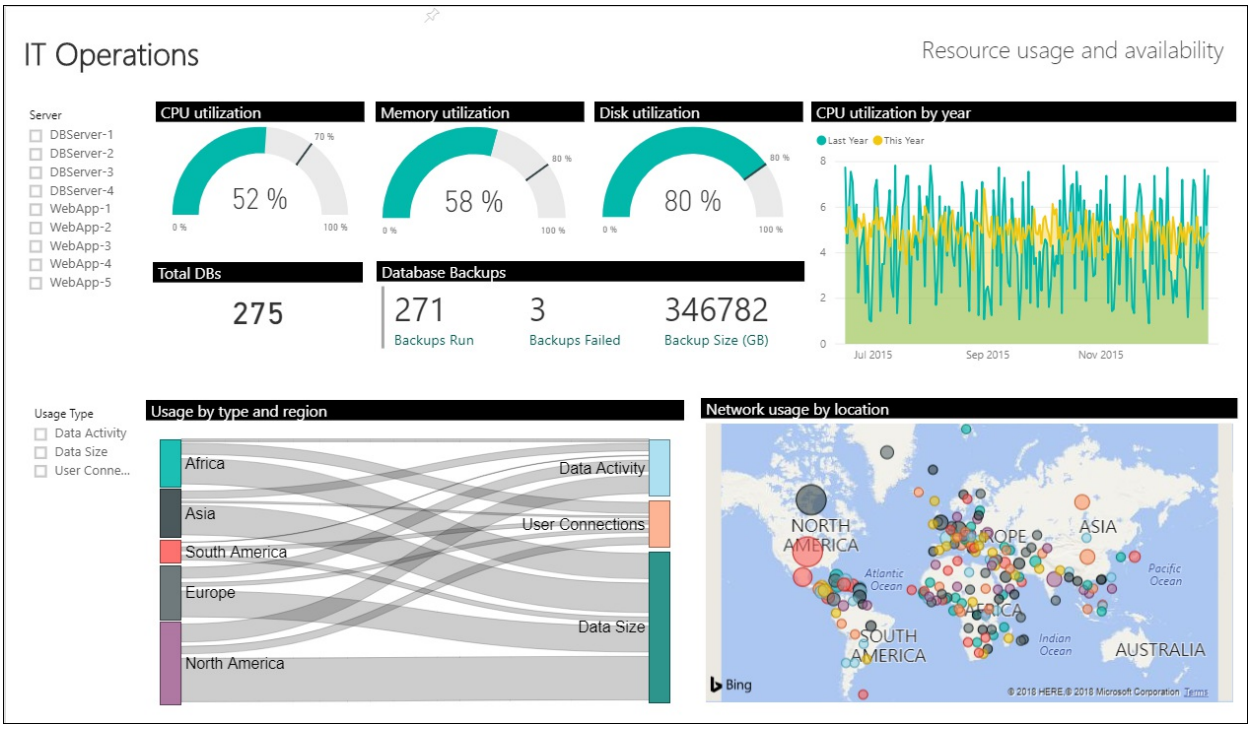
Views

Views visually present log data in Azure Monitor. Each view includes a single tile that drills down to a combination of visualizations such as bar and line charts in addition to lists summarizing critical data. Monitoring solutions include views that summarize data for a particular application, and you can create your own views to present data from any log query. Like other elements in Azure Monitor, views can be added to Azure dashboards.



Power BI

Power BI is a business analytics service that provides interactive visualizations across a variety of data sources and is an effective means of making data available to others within and outside your organization. You can configure Power BI to automatically import log data from Azure Monitor to take advantage of these additional visualizations.



Integrate and export data

You'll often have the requirement to integrate Azure Monitor with other systems and to build custom solutions that use your monitoring data. Other Azure services work with Azure Monitor to provide this integration.

Event Hub

Azure Event Hubs is a streaming platform and event ingestion service that can transform and store data using any real-time analytics provider or batching/storage adapters. Use Event Hubs to stream Azure Monitor data to partner SIEM and monitoring tools.

Logic Apps

Logic Apps is a service that allows you to automate tasks and business processes using workflows that integrate with different systems and services. Activities are available that read and write metrics and logs in Azure Monitor, which allows you to build workflows integrating with a variety of other systems.

API

Multiple APIs are available to read and write metrics and logs to and from Azure Monitor in addition to accessing generated alerts. You can also configure and retrieve alerts. This provides you with essentially unlimited possibilities to build custom solutions that integrate with Azure Monitor.

Next steps

Learn more about:

- Metrics and logs for the data collected by Azure Monitor.
- Data sources for how the different components of your application send telemetry.
- Log queries for analyzing collected data.
- Best practices for monitoring cloud applications and services.

Azure Monitor Frequently Asked Questions

6/5/2020 • 37 minutes to read • Edit Online

This Microsoft FAQ is a list of commonly asked questions about Azure Monitor.

General

What is Azure Monitor?

Azure Monitor is a service in Azure that provides performance and availability monitoring for applications and services in Azure, other cloud environments, or on-premises. Azure Monitor collects data from multiple sources into a common data platform where it can be analyzed for trends and anomalies. Rich features in Azure Monitor assist you in quickly identifying and responding to critical situations that may affect your application.

What's the difference between Azure Monitor, Log Analytics, and Application Insights?

In September 2018, Microsoft combined Azure Monitor, Log Analytics, and Application Insights into a single service to provide powerful end-to-end monitoring of your applications and the components they rely on. Features in Log Analytics and Application Insights have not changed, although some features have been rebranded to Azure Monitor in order to better reflect their new scope. The log data engine and query language of Log Analytics is now referred to as Azure Monitor Logs. See Azure Monitor terminology updates.

What does Azure Monitor cost?

Features of Azure Monitor that are automatically enabled such as collection of metrics and activity logs are provided at no cost. There is a cost associated with other features such as log queries and alerting. See the Azure Monitor pricing page for detailed pricing information.

How do I enable Azure Monitor?

Azure Monitor is enabled the moment that you create a new Azure subscription, and Activity log and platform metrics are automatically collected. Create diagnostic settings to collect more detailed information about the operation of your Azure resources, and add monitoring solutions and insights to provide additional analysis on collected data for particular services.

How do I access Azure Monitor?

Access all Azure Monitor features and data from the **Monitor** menu in the Azure portal. The **Monitoring** section of the menu for different Azure services provides access to the same tools with data filtered to a particular resource. Azure Monitor data is also accessible for a variety of scenarios using CLI, PowerShell, and a REST API.

Is there an on-premises version of Azure Monitor?

No. Azure Monitor is a scalable cloud service that processes and stores large amounts of data, although Azure Monitor can monitor resources that are on-premises and in other clouds.

Can Azure Monitor monitor on-premises resources?

Yes, in addition to collecting monitoring data from Azure resources, Azure Monitor can collect data from virtual machines and applications in other clouds and on-premises. See Sources of monitoring data for Azure Monitor.

Does Azure Monitor integrate with System Center Operations Manager?

You can connect your existing System Center Operations Manager management group to Azure Monitor to collect data from agents into Azure Monitor Logs. This allows you to use log queries and solution to analyze data collected from agents. You can also configure existing System Center Operations Manager agents to send data directly to Azure Monitor. See Connect Operations Manager to Azure Monitor.

What IP addresses does Azure Monitor use?

See IP addresses used by Application Insights and Log Analytics for a listing of the IP addresses and ports required for agents and other external resources to access Azure Monitor.

Monitoring data

Where does Azure Monitor get its data?

Azure Monitor collects data from a variety of sources including logs and metrics from Azure platform and resources, custom applications, and agents running on virtual machines. Other services such as Azure Security Center and Network Watcher collect data into a Log Analytics workspace so it can be analyzed with Azure Monitor data. You can also send custom data to Azure Monitor using the REST API for logs or metrics. See Sources of monitoring data for Azure Monitor.

What data is collected by Azure Monitor?

Azure Monitor collects data from a variety of sources into logs or metrics. Each type of data has its own relative advantages, and each supports a particular set of features in Azure Monitor. There is a single metrics database for each Azure subscription, while you can create multiple Log Analytics workspaces to collect logs depending on your requirements. See Azure Monitor data platform.

Is there a maximum amount of data that I can collect in Azure Monitor?

There is no limit to the amount of metric data you can collect, but this data is stored for a maximum of 93 days. See Retention of Metrics. There is no limit on the amount of log data that you can collect, but it may be affected by the pricing tier you choose for the Log Analytics workspace. See pricing details.

How do I access data collected by Azure Monitor?

Insights and solutions provide a custom experience for working with data stored in Azure Monitor. You can work directly with log data using a log query written in Kusto Query Language (KQL). In the Azure portal, you can write and run queries and interactively analyze data using Log Analytics. Analyze metrics in the Azure portal with the Metrics Explorer. See Analyze log data in Azure Monitor and Getting started with Azure Metrics Explorer.

Solutions and insights

What is an insight in Azure Monitor?

Insights provide a customized monitoring experience for particular Azure services. They use the same metrics and logs as other features in Azure Monitor but may collect additional data and provide a unique experience in the Azure portal. See Insights in Azure Monitor.

To view insights in the Azure portal, see the **Insights** section of the **Monitor** menu or the **Monitoring** section of the service's menu.

What is a solution in Azure Monitor?

Monitoring solutions are packaged sets of logic for monitoring a particular application or service based on Azure Monitor features. They collect log data in Azure Monitor and provide log queries and views for their analysis using a common experience in the Azure portal. See Monitoring solutions in Azure Monitor.

To view solutions in the Azure portal, click **More** in the **Insights** section of the **Monitor** menu. Click **Add** to add additional solutions to the workspace.

Logs

What's the difference between Azure Monitor Logs and Azure Data Explorer?

Azure Data Explorer is a fast and highly scalable data exploration service for log and telemetry data. Azure Monitor Logs is built on top of Azure Data Explorer and uses the same Kusto Query Language (KQL) with some minor

differences. See Azure Monitor log query language differences.

How do I retrieve log data?

All data is retrieved from a Log Analytics workspace using a log query written using Kusto Query Language (KQL). You can write your own queries or use solutions and insights that include log queries for a particular application or service. See Overview of log queries in Azure Monitor.

What is a Log Analytics workspace?

All log data collected by Azure Monitor is stored in a Log Analytics workspace. A workspace is essentially a container where log data is collected from a variety of sources. You may have a single Log Analytics workspace for all your monitoring data or may have requirements for multiple workspaces. See Designing your Azure Monitor Logs deployment.

Can you move an existing Log Analytics workspace to another Azure subscription?

You can move a workspace between resource groups or subscriptions but not to a different region. See Move a Log Analytics workspace to different subscription or resource group.

Why can't I see Query Explorer and Save buttons in Log Analytics?

Query Explorer, **Save** and **New alert rule** buttons are not available when the query scope is set to a specific resource. To create alerts, save or load a query, Log Analytics must be scoped to a workspace. To open Log Analytics in workspace context, select **Logs** from the **Azure Monitor** menu. The last used workspace is selected, but you can select any other workspace. See Log query scope and time range in Azure Monitor Log Analytics

Why am I getting the error: "Register resource provider 'Microsoft.Insights' for this subscription to enable this query" when opening Log Analytics from a VM?

Many resource providers are automatically registered, but you may need to manually register some resource providers. The scope for registration is always the subscription. See Resource providers and types for more information.

Why am I am getting no access error message when opening Log Analytics from a VM?

To view VM Logs, you need to be granted with read permission to the workspaces that stores the VM logs. In these cases, your administrator must grant you with to permissions in Azure.

Metrics

Why are metrics from the guest OS of my Azure virtual machine not showing up in Metrics explorer?

Platform metrics are collected automatically for Azure resources. You must perform some configuration though to collect metrics from the guest OS of a virtual machine. For a Windows VM, install the diagnostic extension and configure the Azure Monitor sink as described in Install and configure Windows Azure diagnostics extension (WAD). For Linux, install the Telegraf agent as described in Collect custom metrics for a Linux VM with the InfluxData Telegraf agent.

Alerts

What is an alert in Azure Monitor?

Alerts proactively notify you when important conditions are found in your monitoring data. They allow you to identify and address issues before the users of your system notice them. There are multiple kinds of alerts:

- Metric - Metric value exceeds a threshold.
- Log query - Results of a log query match defined criteria.
- Activity log - Activity log event matches defined criteria.
- Web test - Results of availability test match defined criteria.

See Overview of alerts in Microsoft Azure.

What is an action group?

An action group is a collection of notifications and actions that can be triggered by an alert. Multiple alerts can use a single action group allowing you to leverage common sets of notifications and actions. See Create and manage action groups in the Azure portal.

What is an action rule?

An action rule allows you to modify the behavior of a set of alerts that match a certain criteria. This allows you to to perform such requirements as disable alert actions during a maintenance window. You can also apply an action group to a set of alerts rather than applying them directly to the alert rules. See Action rules.

Agents

Does Azure Monitor require an agent?

An agent is only required to collect data from the operating system and workloads in virtual machines. The virtual machines can be located in Azure, another cloud environment, or on-premises. See Overview of the Azure Monitor agents.

What's the difference between the Azure Monitor agents?

Azure Diagnostic extension is for Azure virtual machines and collects data to Azure Monitor Metrics, Azure Storage, and Azure Event Hubs. The Log Analytics agent is for virtual machines in Azure, another cloud environment, or on-premises and collects data to Azure Monitor Logs. The Dependency agent requires the Log Analytics agent and collected process details and dependencies. See Overview of the Azure Monitor agents.

Does my agent traffic use my ExpressRoute connection?

Traffic to Azure Monitor uses the Microsoft peering ExpressRoute circuit. See ExpressRoute documentation for a description of the different types of ExpressRoute traffic.

How can I confirm that the Log Analytics agent is able to communicate with Azure Monitor?

From Control Panel on the agent computer, select **Security & Settings**, **Microsoft Monitoring Agent** . Under the **Azure Log Analytics (OMS)** tab, a green check mark icon confirms that the agent is able to communicate with Azure Monitor. A yellow warning icon means the agent is having issues. One common reason is the **Microsoft Monitoring Agent** service has stopped. Use service control manager to restart the service.

How do I stop the Log Analytics agent from communicating with Azure Monitor?

For agents connected to Log Analytics directly, open the Control Panel and select **Security & Settings**, **Microsoft Monitoring Agent**. Under the **Azure Log Analytics (OMS)** tab, remove all workspaces listed. In System Center Operations Manager, remove the computer from the Log Analytics managed computers list. Operations Manager updates the configuration of the agent to no longer report to Log Analytics.

How much data is sent per agent?

The amount of data sent per agent depends on:

- The solutions you have enabled
- The number of logs and performance counters being collected
- The volume of data in the logs

See Manage usage and costs with Azure Monitor Logs for details.

For computers that are able to run the WireData agent, use the following query to see how much data is being sent:

```
WireData
| where ProcessName == "C:\\Program Files\\Microsoft Monitoring Agent\\Agent\\MonitoringHost.exe"
| where Direction == "Outbound"
| summarize sum(TotalBytes) by Computer
```

How much network bandwidth is used by the Microsoft Management Agent (MMA) when sending data to Azure Monitor?

Bandwidth is a function on the amount of data sent. Data is compressed as it is sent over the network.

How can I be notified when data collection from the Log Analytics agent stops?

Use the steps described in create a new log alert to be notified when data collection stops. Use the following settings for the alert rule:

- **Define alert condition**: Specify your Log Analytics workspace as the resource target.
- **Alert criteria**
  - **Signal Name**: *Custom log search*
  - **Search query**:
    ```
    Heartbeat | summarize LastCall = max(TimeGenerated) by Computer | where LastCall < ago(15m)
    ```
  - **Alert logic**: **Based on** *number of results*, **Condition** *Greater than*, **Threshold value** *0*
  - **Evaluated based on**: **Period (in minutes)** *30*, **Frequency (in minutes)** *10*
- **Define alert details**
  - **Name**: *Data collection stopped*
  - **Severity**: *Warning*

Specify an existing or new Action Group so that when the log alert matches criteria, you are notified if you have a heartbeat missing for more than 15 minutes.

What are the firewall requirements for Azure Monitor agents?

See Network firewall requirementsfor details on firewall requirements.

Visualizations

Why can't I see View Designer?

View Designer is only available for users assigned with Contributor permissions or higher in the Log Analytics workspace.

Application Insights

Configuration problems

I'm having trouble setting up my:

- .NET app
- Monitoring an already-running app
- Azure diagnostics
- Java web app

I get no data from my server:

- Set firewall exceptions
- Set up an ASP.NET server
- Set up a Java server

How many Application Insights should I deploy?:

- How to design your Application Insights deployment: One versus many Application Insights resources?

Can I use Application Insights with ...?

- Web apps on an IIS server in Azure VM or Azure virtual machine scale set
- Web apps on an IIS server - on-premises or in a VM

- Java web apps
- Node.js apps
- Web apps on Azure
- Cloud Services on Azure
- App servers running in Docker
- Single-page web apps
- SharePoint
- Windows desktop app
- Other platforms

Is it free?

Yes, for experimental use. In the basic pricing plan, your application can send a certain allowance of data each month free of charge. The free allowance is large enough to cover development, and publishing an app for a small number of users. You can set a cap to prevent more than a specified amount of data from being processed.

Larger volumes of telemetry are charged by the Gb. We provide some tips on how to limit your charges.

The Enterprise plan incurs a charge for each day that each web server node sends telemetry. It is suitable if you want to use Continuous Export on a large scale.

Read the pricing plan.

How much does it cost?

- Open the **Usage and estimated costs page** in an Application Insights resource. There's a chart of recent usage. You can set a data volume cap, if you want.
- Open the Azure Billing blade to see your bills across all resources.

What does Application Insights modify in my project?

The details depend on the type of project. For a web application:

- Adds these files to your project:
  - ApplicationInsights.config
  - ai.js
- Installs these NuGet packages:
  - *Application Insights API* - the core API
  - *Application Insights API for Web Applications* - used to send telemetry from the server
  - *Application Insights API for JavaScript Applications* - used to send telemetry from the client
- The packages include these assemblies:
  - Microsoft.ApplicationInsights
  - Microsoft.ApplicationInsights.Platform
- Inserts items into:
  - Web.config
  - packages.config
- (New projects only - if you add Application Insights to an existing project, you have to do this manually.) Inserts snippets into the client and server code to initialize them with the Application Insights resource ID. For example, in an MVC app, code is inserted into the master page Views/Shared/_Layout.cshtml

How do I upgrade from older SDK versions?

See the release notes for the SDK appropriate to your type of application.

How can I change which Azure resource my project sends data to?

In Solution Explorer, right-click `ApplicationInsights.config` and choose **Update Application Insights**. You can

send the data to an existing or new resource in Azure. The update wizard changes the instrumentation key in ApplicationInsights.config, which determines where the server SDK sends your data. Unless you deselect "Update all," it will also change the key where it appears in your web pages.

Can I use `providers('Microsoft.Insights', 'components').apiVersions[0]` in my Azure Resource Manager deployments?

We do not recommend using this method of populating the API version. The newest version can represent preview releases which may contain breaking changes. Even with newer non-preview releases, the API versions are not always backwards compatible with existing templates, or in some cases the API version may not be available to all subscriptions.

What is Status Monitor?

A desktop app that you can use in your IIS web server to help configure Application Insights in web apps. It doesn't collect telemetry: you can stop it when you are not configuring an app.

Learn more.

What telemetry is collected by Application Insights?

From server web apps:

- HTTP requests
- Dependencies. Calls to: SQL Databases; HTTP calls to external services; Azure Cosmos DB, table, blob storage, and queue.
- Exceptions and stack traces.
- Performance Counters - If you use Status Monitor, Azure monitoring for App Services, Azure monitoring for VM or virtual machine scale set, or the Application Insights collectd writer.
- Custom events and metrics that you code.
- Trace logs if you configure the appropriate collector.

From client web pages:

- Page view counts
- AJAX calls Requests made from a running script.
- Page view load data
- User and session counts
- Authenticated user IDs

From other sources, if you configure them:

- Azure diagnostics
- Import to Analytics
- Log Analytics
- Logstash

Can I filter out or modify some telemetry?

Yes, in the server you can write:

- Telemetry Processor to filter or add properties to selected telemetry items before they are sent from your app.
- Telemetry Initializer to add properties to all items of telemetry.

Learn more for ASP.NET or Java.

How are city, country/region, and other geo location data calculated?

We look up the IP address (IPv4 or IPv6) of the web client using GeoLite2.

- Browser telemetry: We collect the sender's IP address.
- Server telemetry: The Application Insights module collects the client IP address. It is not collected if `X-Forwarded-For` is set.
- To learn more about how IP address and geolocation data is collected in Application Insights refer to this article.

You can configure the `ClientIpHeaderTelemetryInitializer` to take the IP address from a different header. In some systems, for example, it is moved by a proxy, load balancer, or CDN to `X-Originating-IP`. Learn more.

You can use Power BI to display your request telemetry on a map.

How long is data retained in the portal? Is it secure?

Take a look at Data Retention and Privacy.

What happens to Application Insight's telemetry when a server or device loses connection with Azure?

All of our SDKs, including the web SDK, includes "reliable transport" or "robust transport". When the server or device loses connection with Azure, telemetry is stored locally on the file system (Server SDKs) or in HTML5 Session Storage (Web SDK). The SDK will periodically retry to send this telemetry until our ingestion service considers it "stale" (48-hours for logs, 30 minutes for metrics). Stale telemetry will be dropped. In some cases, such as when local storage is full, retry will not occur.

Could personal data be sent in the telemetry?

This is possible if your code sends such data. It can also happen if variables in stack traces include personal data. Your development team should conduct risk assessments to ensure that personal data is properly handled. Learn more about data retention and privacy.

All octets of the client web address are always set to 0 after the geo location attributes are looked up.

My Instrumentation Key is visible in my web page source.

- This is common practice in monitoring solutions.
- It can't be used to steal your data.
- It could be used to skew your data or trigger alerts.
- We have not heard that any customer has had such problems.

You could:

- Use two separate Instrumentation Keys (separate Application Insights resources), for client and server data. Or
- Write a proxy that runs in your server, and have the web client send data through that proxy.

How do I see POST data in Diagnostic search?

We don't log POST data automatically, but you can use a TrackTrace call: put the data in the message parameter. This has a longer size limit than the limits on string properties, though you can't filter on it.

Should I use single or multiple Application Insights resources?

Use a single resource for all the components or roles in a single business system. Use separate resources for development, test, and release versions, and for independent applications.

- See the discussion here
- Example - cloud service with worker and web roles

How do I dynamically change the instrumentation key?

- Discussion here
- Example - cloud service with worker and web roles

What are the User and Session counts?

- The JavaScript SDK sets a user cookie on the web client, to identify returning users, and a session cookie to

group activities.

- If there is no client-side script, you can set cookies at the server.
- If one real user uses your site in different browsers, or using in-private/incognito browsing, or different machines, then they will be counted more than once.
- To identify a logged-in user across machines and browsers, add a call to setAuthenticatedUserContext().

Have I enabled everything in Application Insights?

WHAT YOU SHOULD SEE	HOW TO GET IT	WHY YOU WANT IT
Availability charts	Web tests	Know your web app is up
Server app perf: response times, ...	Add Application Insights to your project or Install AI Status Monitor on server (or write your own code to track dependencies)	Detect perf issues
Dependency telemetry	Install AI Status Monitor on server	Diagnose issues with databases or other external components
Get stack traces from exceptions	Insert TrackException calls in your code (but some are reported automatically)	Detect and diagnose exceptions
Search log traces	Add a logging adapter	Diagnose exceptions, perf issues
Client usage basics: page views, sessions, ...	JavaScript initializer in web pages	Usage analytics
Client custom metrics	Tracking calls in web pages	Enhance user experience
Server custom metrics	Tracking calls in server	Business intelligence

Why are the counts in Search and Metrics charts unequal?

Sampling reduces the number of telemetry items (requests, custom events, and so on) that are actually sent from your app to the portal. In Search, you see the number of items actually received. In metric charts that display a count of events, you see the number of original events that occurred.

Each item that is transmitted carries an `itemCount` property that shows how many original events that item represents. To observe sampling in operation, you can run this query in Analytics:

```
requests | summarize original_events = sum(itemCount), transmitted_events = count()
```

Automation

Configuring Application Insights

You can write PowerShell scripts using Azure Resource Monitor to:

- Create and update Application Insights resources.
- Set the pricing plan.
- Get the instrumentation key.
- Add a metric alert.
- Add an availability test.

You can't set up a Metric Explorer report or set up continuous export.

Querying the telemetry

Use the REST API to run Analytics queries.

How can I set an alert on an event?

Azure alerts are only on metrics. Create a custom metric that crosses a value threshold whenever your event occurs. Then set an alert on the metric. You'll get a notification whenever the metric crosses the threshold in either direction; you won't get a notification until the first crossing, no matter whether the initial value is high or low; there is always a latency of a few minutes.

Are there data transfer charges between an Azure web app and Application Insights?

- If your Azure web app is hosted in a data center where there is an Application Insights collection endpoint, there is no charge.
- If there is no collection endpoint in your host data center, then your app's telemetry will incur Azure outgoing charges.

This doesn't depend on where your Application Insights resource is hosted. It just depends on the distribution of our endpoints.

Can I send telemetry to the Application Insights portal?

We recommend you use our SDKs and use the SDK API. There are variants of the SDK for various platforms. These SDKs handle buffering, compression, throttling, retries, and so on. However, the ingestion schema and endpoint protocol are public.

Can I monitor an intranet web server?

Yes, but you will need to allow traffic to our services by either firewall exceptions or proxy redirects.

- QuickPulse `https://rt.services.visualstudio.com:443`
- ApplicationIdProvider `https://dc.services.visualstudio.com:443`
- TelemetryChannel `https://dc.services.visualstudio.com:443`

Review our full list of services and IP addresses here.

Firewall exception

Allow your web server to send telemetry to our endpoints.

Gateway redirect

Route traffic from your server to a gateway on your intranet by overwriting Endpoints in your configuration. If these "Endpoint" properties are not present in your config, these classes will use the default values shown below in the example ApplicationInsights.config.

Your gateway should route traffic to our endpoint's base address. In your configuration, replace the default values with `http://<your.gateway.address>/<relative path>`.

Example ApplicationInsights.config with default endpoints:

```
<ApplicationInsights>
  ...
  <TelemetryModules>
    <Add
Type="Microsoft.ApplicationInsights.Extensibility.PerfCounterCollector.QuickPulse.QuickPulseTelemetryModule,
Microsoft.AI.PerfCounterCollector">

<QuickPulseServiceEndpoint>https://rt.services.visualstudio.com/QuickPulseService.svc</QuickPulseServiceEndpoin
t>
    </Add>
  </TelemetryModules>
    ...
  <TelemetryChannel>
     <EndpointAddress>https://dc.services.visualstudio.com/v2/track</EndpointAddress>
  </TelemetryChannel>
  ...
  <ApplicationIdProvider
Type="Microsoft.ApplicationInsights.Extensibility.Implementation.ApplicationId.ApplicationInsightsApplicationId
Provider, Microsoft.ApplicationInsights">
    <ProfileQueryEndpoint>https://dc.services.visualstudio.com/api/profiles/{0}/appId</ProfileQueryEndpoint>
  </ApplicationIdProvider>
  ...
</ApplicationInsights>
```

NOTE

ApplicationIdProvider is available starting in v2.6.0.

Proxy passthrough

Proxy passthrough can be achieved by configuring either a machine level or application level proxy. For more information see dotnet's article on DefaultProxy.

Example Web.config:

```
<system.net>
   <defaultProxy>
     <proxy proxyaddress="http://xx.xx.xx.xx:yyyy" bypassonlocal="true"/>
   </defaultProxy>
</system.net>
```

Can I run Availability web tests on an intranet server?

Our web tests run on points of presence that are distributed around the globe. There are two solutions:

- Firewall door - Allow requests to your server from the long and changeable list of web test agents.
- Write your own code to send periodic requests to your server from inside your intranet. You could run Visual Studio web tests for this purpose. The tester could send the results to Application Insights using the TrackAvailability() API.

How long does it take for telemetry to be collected?

Most Application Insights data has a latency of under 5 minutes. Some data can take longer; typically larger log files. For more information, see the Application Insights SLA.

Azure Monitor for containers

This Microsoft FAQ is a list of commonly asked questions about Azure Monitor for containers. If you have any additional questions about the solution, go to the discussion forum and post your questions. When a question is frequently asked, we add it to this article so that it can be found quickly and easily.

Health feature is in private preview

We are planning to make a series of changes to add functionality and address your feedback. The Health feature is going to transition to a private preview at the end of June 2020, and for additional information review the following Azure updates announcement.

What does *Other Processes* represent under the Node view?

Other processes is intended to help you clearly understand the root cause of the high resource usage on your node. This enables you to distinguish usage between containerized processes vs non-containerized processes.

What are these **Other Processes**?

These are non-containerized processes that run on your node.

How do we calculate this?

Other Processes = *Total usage from CAdvisor - Usage from containerized process*

The **Other processes** includes:

- Self-managed or managed Kubernetes non-containerized processes
- Container Run-time processes
- Kubelet
- System processes running on your node
- Other non-Kubernetes workloads running on node hardware or VM

I don't see Image and Name property values populated when I query the ContainerLog table.

For agent version ciprod12042019 and later, by default these two properties are not populated for every log line to minimize cost incurred on log data collected. There are two options to query the table that include these properties with their values:

Option 1

Join other tables to include these property values in the results.

Modify your queries to include Image and ImageTag properties from the `ContainerInventory` table by joining on ContainerID property. You can include the Name property (as it previously appeared in the `ContainerLog` table) from KubepodInventory table's ContaineName field by joining on the ContainerID property.This is the recommended option.

The following example is a sample detailed query that explains how to get these field values with joins.

```
//lets say we are querying an hour worth of logs
let startTime = ago(1h);
let endTime = now();
//below gets the latest Image & ImageTag for every containerID, during the time window
let ContainerInv = ContainerInventory | where TimeGenerated >= startTime and TimeGenerated < endTime |
summarize arg_max(TimeGenerated, *)  by ContainerID, Image, ImageTag | project-away TimeGenerated | project
ContainerID1=ContainerID, Image1=Image ,ImageTag1=ImageTag;
//below gets the latest Name for every containerID, during the time window
let KubePodInv  = KubePodInventory | where ContainerID != "" | where TimeGenerated >= startTime | where
TimeGenerated < endTime | summarize arg_max(TimeGenerated, *)  by ContainerID2 = ContainerID,
Name1=ContainerName | project ContainerID2 , Name1;
//now join the above 2 to get a 'jointed table' that has name, image & imagetag. Outer left is safer in-case
there are no kubepod records are if they are latent
let ContainerData = ContainerInv | join kind=leftouter (KubePodInv) on $left.ContainerID1 ==
$right.ContainerID2;
//now join ContainerLog table with the 'jointed table' above and project-away redundant fields/columns and
rename columns that were re-written
//Outer left is safer so you dont lose logs even if we cannot find container metadata for loglines (due to
latency, time skew between data types etc...)
ContainerLog
| where TimeGenerated >= startTime and TimeGenerated < endTime
| join kind= leftouter (
   ContainerData
) on $left.ContainerID == $right.ContainerID2 | project-away ContainerID1, ContainerID2, Name, Image, ImageTag
| project-rename Name = Name1, Image=Image1, ImageTag=ImageTag1
```

Option 2

Re-enable collection for these properties for every container log line.

If the first option is not convenient due to query changes involved, you can re-enable collecting these fields by enabling the setting `log_collection_settings.enrich_container_logs` in the agent config map as described in the data collection configuration settings.

> **NOTE**
> The second option is not recommended with large clusters that have more than 50 nodes because it generates API server calls from every node in the cluster to perform this enrichment. This option also increases data size for every log line collected.

Can I view metrics collected in Grafana?

Azure Monitor for containers supports viewing metrics stored in your Log Analytics workspace in Grafana dashboards. We have provided a template that you can download from Grafana's dashboard repository to get you started and reference to help you learn how to query additional data from your monitored clusters to visualize in custom Grafana dashboards.

Can I monitor my AKS-engine cluster with Azure Monitor for containers?

Azure Monitor for containers supports monitoring container workloads deployed to AKS-engine (formerly known as ACS-engine) cluster(s) hosted on Azure. For further details and an overview of steps required to enable monitoring for this scenario, see Using Azure Monitor for containers for AKS-engine.

Why don't I see data in my Log Analytics workspace?

If you are unable to see any data in the Log Analytics workspace at a certain time everyday, you may have reached the default 500 MB limit or the daily cap specified to control the amount of data to collect daily. When the limit is met for the day, data collection stops and resumes only on the next day. To review your data usage and update to a different pricing tier based on your anticipated usage patterns, see Log data usage and cost.

What are the container states specified in the ContainerInventory table?

The ContainerInventory table contains information about both stopped and running containers. The table is

populated by a workflow inside the agent that queries the docker for all the containers (running and stopped), and forwards that data the Log Analytics workspace.

How do I resolve *Missing Subscription registration* error?

If you receive the error **Missing Subscription registration for Microsoft.OperationsManagement**, you can resolve it by registering the resource provider **Microsoft.OperationsManagement** in the subscription where the workspace is defined. The documentation for how to do this can be found here.

Is there support for RBAC enabled AKS clusters?

The Container Monitoring solution doesn't support RBAC, but it is supported with Azure Monitor for Containers. The solution details page may not show the right information in the blades that show data for these clusters.

How do I enable log collection for containers in the kube-system namespace through Helm?

The log collection from containers in the kube-system namespace is disabled by default. Log collection can be enabled by setting an environment variable on the omsagent. For more information, see the Azure Monitor for containers GitHub page.

How do I update the omsagent to the latest released version?

To learn how to upgrade the agent, see Agent management.

How do I enable multi-line logging?

Currently Azure Monitor for containers doesn't support multi-line logging, but there are workarounds available. You can configure all the services to write in JSON format and then Docker/Moby will write them as a single line.

For example, you can wrap your log as a JSON object as shown in the example below for a sample node.js application:

```
console.log(json.stringify({
      "Hello": "This example has multiple lines:",
      "Docker/Moby": "will not break this into multiple lines",
      "and you will receive":"all of them in log analytics",
      "as one": "log entry"
      }));
```

This data will look like the following example in Azure Monitor for logs when you query for it:

```
LogEntry : ({"Hello": "This example has multiple lines:","Docker/Moby": "will not break this into multiple
lines", "and you will receive":"all of them in log analytics", "as one": "log entry"}
```

For a detailed look at the issue, review the following GitHub link.

How do I resolve Azure AD errors when I enable live logs?

You may see the following error: **The reply url specified in the request does not match the reply urls configured for the application: '<application ID>'**. The solution to solve it can be found in the article How to view container data in real time with Azure Monitor for containers.

Why can't I upgrade cluster after onboarding?

If after you enable Azure Monitor for containers for an AKS cluster, you delete the Log Analytics workspace the cluster was sending its data to, when attempting to upgrade the cluster it will fail. To work around this, you will have to disable monitoring and then re-enable it referencing a different valid workspace in your subscription. When you try to perform the cluster upgrade again, it should process and complete successfully.

Which ports and domains do I need to open/whitelist for the agent?

See the Network firewall requirements for the proxy and firewall configuration information required for the

containerized agent with Azure, Azure US Government, and Azure China 21Vianet clouds.

Azure Monitor for VMs

This Microsoft FAQ is a list of commonly asked questions about Azure Monitor for VMs. If you have any additional questions about the solution, go to the discussion forum and post your questions. When a question is frequently asked, we add it to this article so that it can be found quickly and easily.

Can I onboard to an existing workspace?

If your virtual machines are already connected to a Log Analytics workspace, you may continue to use that workspace when onboarding to Azure Monitor for VMs, provided it is in one of the supported regions listed here.

Can I onboard to a new workspace?

If your VMs are not currently connected to an existing Log Analytics workspace, you need to create a new workspace to store your data. Creating a new default workspace is done automatically if you configure a single Azure VM for Azure Monitor for VMs through the Azure portal.

If you choose to use the script-based method, these steps are covered in the Enable Azure Monitor for VMs using Azure PowerShell or Resource Manager template article.

What do I do if my VM is already reporting to an existing workspace?

If you are already collecting data from your virtual machines, you may have already configured it to report data to an existing Log Analytics workspace. As long as that workspace is in one of our supported regions, you can enable Azure Monitor for VMs to that pre-existing workspace. If the workspace you are already using is not in one of our supported regions, you won't be able to onboard to Azure Monitor for VMs at this time. We are actively working to support additional regions.

Why did my VM fail to onboard?

When onboarding an Azure VM from the Azure portal, the following steps occur:

- A default Log Analytics workspace is created, if that option was selected.
- The Log Analytics agent is installed on Azure VMs using a VM extension, if determined it is required.
- The Azure Monitor for VMs Map Dependency agent is installed on Azure VMs using an extension, if determined it is required.

During the onboard process, we check for status on each of the above to return a notification status to you in the portal. Configuration of the workspace and the agent installation typically takes 5 to 10 minutes. Viewing monitoring data in the portal take an additional 5 to 10 minutes.

If you have initiated onboarding and see messages indicating the VM needs to be onboarded, allow for up to 30 minutes for the VM to complete the process.

I don't see some or any data in the performance charts for my VM

Our performance charts have been updated to use data stored in the *InsightsMetrics* table. To see data in these charts you will need to upgrade to use the new VM Insights solution. Please refer to our GA FAQ for additional information.

If you don't see performance data in the disk table or in some of the performance charts then your performance counters may not be configured in the workspace. To resolve, run the following PowerShell script.

How is Azure Monitor for VMs Map feature different from Service Map?

The Azure Monitor for VMs Map feature is based on Service Map, but has the following differences:

- The Map view can be accessed from the VM blade and from Azure Monitor for VMs under Azure Monitor.
- The connections in the Map are now clickable and display a view of the connection metric data in the side panel for the selected connection.

- There is a new API that is used to create the maps to better support more complex maps.
- Monitored VMs are now included in the client group node, and the donut chart shows the proportion of monitored vs unmonitored virtual machines in the group. It can also be used to filter the list of machines when the group is expanded.
- Monitored virtual machines are now included in the server port group nodes, and the donut chart shows the proportion of monitored vs unmonitored machines in the group. It can also be used to filter the list of machines when the group is expanded.
- The map style has been updated to be more consistent with App Map from Application insights.
- The side panels have been updated, and do not have the full set of integration's that were supported in Service Map - Update Management, Change Tracking, Security, and Service Desk.
- The option for choosing groups and machines to map has been updated and now supports Subscriptions, Resource Groups, Azure virtual machine scale sets, and Cloud services.
- You cannot create new Service Map machine groups in the Azure Monitor for VMs Map feature.

Why do my performance charts show dotted lines?

This can occur for a few reasons. In cases where there is a gap in data collection we depict the lines as dotted. If you have modified the data sampling frequency for the performance counters enabled (the default setting is to collect data every 60 seconds), you can see dotted lines in the chart if you choose a narrow time range for the chart and your sampling frequency is less than the bucket size used in the chart (for example, the sampling frequency is every 10 minutes and each bucket on the chart is 5 minutes). Choosing a wider time range to view should cause the chart lines to appear as solid lines rather than dots in this case.

Are groups supported with Azure Monitor for VMs?

Yes, once you install the Dependency agent we collect information from the VMs to display groups based upon subscription, resource group, virtual machine scale sets, and cloud services. If you have been using Service Map and have created machine groups, these are displayed as well. Computer groups will also appear in the groups filter if you have created them for the workspace you are viewing.

How do I see the details for what is driving the 95th percentile line in the aggregate performance charts?

By default, the list is sorted to show you the VMs that have the highest value for the 95th percentile for the selected metric, except for the Available memory chart, which shows the machines with the lowest value of the 5th percentile. Clicking on the chart will open the **Top N List** view with the appropriate metric selected.

How does the Map feature handle duplicate IPs across different vnets and subnets?

If you are duplicating IP ranges either with VMs or Azure virtual machine scale sets across subnets and vnets, it can cause Azure Monitor for VMs Map to display incorrect information. This is a known issue and we are investigating options to improve this experience.

Does Map feature support IPv6?

Map feature currently only supports IPv4 and we are investigating support for IPv6. We also support IPv4 that is tunneled inside IPv6.

When I load a map for a Resource Group or other large group the map is difficult to view

While we have made improvements to Map to handle large and complex configurations, we realize a map can have a lot of nodes, connections, and node working as a cluster. We are committed to continuing to enhance support to increase scalability.

Why does the network chart on the Performance tab look different than the network chart on the Azure VM Overview page?

The overview page for an Azure VM displays charts based on the host's measurement of activity in the guest VM. For the network chart on the Azure VM Overview, it only displays network traffic that will be billed. This does not include inter-virtual network traffic. The data and charts shown for Azure Monitor for VMs is based on data from the guest VM and the network chart displays all TCP/IP traffic that is inbound and outbound to that VM, including

inter-virtual network.

How is response time measured for data stored in VMConnection and displayed in the connection panel and workbooks?

Response time is an approximation. Since we do not instrument the code of the application, we do not really know when a request begins and when the response arrives. Instead we observe data being sent on a connection and then data coming back on that connection. Our agent keeps track of these sends and receives and attempts to pair them: a sequence of sends, followed by a sequence of receives is interpreted as a request/response pair. The timing between these operations is the response time. It will include the network latency and the server processing time.

This approximation works well for protocols that are request/response based: a single request goes out on the connection, and a single response arrives. This is the case for HTTP(S) (without pipelining), but not satisfied for other protocols.

Are their limitations if I am on the Log Analytics Free pricing plan?

If you have configured Azure Monitor with a Log Analytics workspace using the *Free* pricing tier, Azure Monitor for VMs Map feature will only support five connected machines connected to the workspace. If you have five VMs connected to a free workspace, you disconnect one of the VMs and then later connect a new VM, the new VM is not monitored and reflected on the Map page.

Under this condition, you will be prompted with the **Try Now** option when you open the VM and select **Insights** from the left-hand pane, even after it has been installed already on the VM. However, you are not prompted with options as would normally occur if this VM were not onboarded to Azure Monitor for VMs.

Next steps

If your question isn't answered here, you can refer to the following forums to additional questions and answers.

- Log Analytics
- Application Insights

For general feedback on Azure Monitor please visit the feedback forum.